UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/774,236	01/29/2001	Daniel Isaac Goodman	FIN0004-CIP4	9845

74877 7590 03/17/2009

King and Spalding LLP
1700 Pennsylvania Ave, NW
Suite 200
Washington, DC 20006

EXAMINER	
KHOSHNOODI, NADIA	
ART UNIT	PAPER NUMBER
2437	

MAIL DATE	DELIVERY MODE
03/17/2009	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	09/774,236	GOODMAN ET AL.	
	Examiner	Art Unit	
	NADIA KHOSHNOODI	2437	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *19 November 2008*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *See Continuation Sheet* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *See Continuation Sheet* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *29 January 2001* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date *4/18/2008*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

Continuation of Disposition of Claims: Claims pending in the application are 1-12,14-18,25-37,39,40,42,43,50-58,60,62,63,69-79,81,83,84,88-92,115-132,141,142 and 171-174.

Continuation of Disposition of Claims: Claims rejected are 1-12,14-18,25-37,39,40,42,43,50-58,60,62,63,69-79,81,83,84,88-92,115-132,141,142 and 171-174.

DETAILED ACTION

Response to Amendment

Claims 1-12, 14-18, 25-37, 39-40, 42-43, 50-58, 60, 62-63, 69-79, 81, 83-84, 88-92, 115-132, 141-142, 171-174 are pending. Applicant's arguments/amendments with respect to the pending claims filed 11/19/2008 have been fully considered but they are not persuasive. The Examiner would like to point out that this action is made final (See MPEP 706.07a).

Response to Arguments

Applicants contend that "neither Pomerantz nor Saito discloses decrypting encrypted text within a patched operating system function." Examiner respectfully disagrees. Saito teaches that the text string information is displayed using a TextOut() and ExtTextOut() function in an API, i.e. using a patched operating system function (col. 5, lines 28-39). Specifically, in that citing Saito also teaches using a hook to go to a function different than it originally would if the text extraction system had not been activated. An API is commonly known as being an interface that is used by an application to gain access to the operating system. Furthermore, Applicants define various ways to meet the criteria of patching as intervening with a function call in par. 84 of the disclosure, these techniques including: (1) inserting additional instructions into the function, (2) re-directing the call to the function to that of different function, or (3) changing the address of the function within a look-up table to an address of a different function. Thus, in view of Applicant's definition and the fact that Saito discloses that the API hooks the text function to take text from the appropriate area (col. 5, lines 28-39), Saito teaches the use of a patched operating system function. Saito was then used in combination with Pomerantz et al. to perform the decryption of

encrypted text within a patched operating system function. Thus, the combination of Saito and Pomerantz teaches/suggests decrypting encrypted text within a patched operating system function.

Due to the reasons stated above, the Examiner maintains the prior art rejections with respect to the pending claims. The cited prior arts of record taken in combination with one another teach the limitations that the Applicant suggests distinguish from the prior art, since the claims are given their broadest reasonable interpretation (MPEP 2111). Therefore, it is the Examiner's conclusion that the pending claims are not patentably distinct or non-obvious over the prior art of record as presented.

Claim Objections

Claims 51 and 172 are objected to because of the following informalities: these claims as amended recite "…wherein the include at least one…" Appropriate correction is required.

Claim Rejections - 35 USC § 112

I. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

II. Claim 43 contains the trademark/trade names "Macintosh DrawText." Where a trademark or trade name is used in a claim as a limitation to identify or describe a particular material or product, the claim does not comply with the requirements of 35 U.S.C. 112, second paragraph. *SeeExparte Simpson,* 218 USPQ 1020 (Bd. App. 1982). The claim scope is uncertain since the trademark or trade name cannot be used properly to identify any particular material or

product. A trademark or trade name is used to identify a source of goods, and not the goods themselves. Thus, a trademark or trade name does not identify or describe the goods associated with the trademark or trade name. In the present case, the trademark/trade name is used to identify/describe a specific operating system function and, accordingly, the identification/description is indefinite. For purposes of examination, the trademarked portion of the claim limitation will not be considered.

Claim Rejections - 35 USC § 103

III. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

IV. Claims 1-3, 5-8, 12, 14-18, 25-28, 30-33, 37, 39-40, 43, 50-53, 55-58, 60, 62-63, 69-74, 76-79, 81, 83-84, 88-92, 115-132, 141-142, and 171-174 are rejected under 35 U.S.C. 103(a) as being unpatentable over Pomerantz et al. US Patent No. 6,178,243 and further in view of Saito et al., US Patent No. 5,900,005.

As per claims 1, 26, and 171:

Pomerantz et al. substantially teach a method/system/computer readable medium with stored program code for protecting content within a page displayed by a computer, comprising identifying a designated portion of original content contained within the page (col. 6, lines 42-54 and Fig. 2A), modifying the page comprising encrypting the designated portion of original content to form a portion of encrypted content (col. 6, lines 42-54 and Fig. 2A), replacing the

designated portion of original content within the page with the portion of encrypted content is (col. 7, lines 1-9 and Fig. 2B), rendering[1] the page into a graphics device; and converting the page into output data for a graphics device (col. 6, lines 28-34); and displaying the output data (col. 6, lines 28-34).

Not explicitly disclosed is wherein rendering the page includes dynamically generating a layout for display of the page based on spatial characteristics of the decrypted text wherein the spatial characteristics include (j) numbers of words per line and decrypting the portion of encrypted text within a patched operating system function to produce decrypted text. However, Saito teaches that when rendering a display to a web page, an operating system function is used to replace words/text strings with different text, where the operating system has a patched operative system function call to the text that is to be output (col. 5, lines 28-39 and col. 6, line 1-7). Therefore, it would have been obvious to a person in the art at the time the invention was made to modify the method disclosed in Pomerantz et al. for the rendering function to include dynamically generating a display layout for a web site and replacing the text input portion, which would be the on-screen section selected for a cut/copy command (taught by Pomerantz et al.), with another text output portion which is the decrypted form of the text retrieved by the patched operating system function such as TextOut() or ExtTextOut() function (taught by Saito) and then displaying the decrypted text to the screen. This modification would have been obvious because a person having ordinary skill in the art, at the time the invention was made, would have been

[1] Although the word "render" was not specifically used, the definition according to www.netlingo.com shows that the function of rendering does take place. Below is the definition of render used.

> **Render** - To depict something. For example, an HTML author creatively renders text and graphics on a Web page into columns and rows, and a browser automatically renders the Web page by interpreting the HTML code.

motivated to do so since Saito teaches that rendering text in the format described allows for convenient text extraction and replacing any arbitrary word(s) on a screen in col. 1, lines 59-61.

As per claims 2, 27, 52, and 73:

Pomerantz et al. and Saito substantially teach the method/system as applied to claims 1, 26, 51, and 72. Furthermore, Saito teaches the method/system wherein the page is a web page (col. 4, lines 35-44).

As per claims 3, 28, 53, and 74:

Pomerantz et al. and Saito substantially teach the method/system as applied to claims 2, 27, 52, and 73. Furthermore, Saito teaches the method/system wherein the web page is an HTML page (col. 4, lines 35-44).

As per claims 5, 30, 55, and 76:

Pomerantz et al. and Saito substantially teach the method/system as applied to claims 1, 26, 51, and 72. Furthermore, Saito teaches the method/system wherein the page is part of a document produced by a software application (col. 8, lines 56-60).

As per claims 6, 31, 56, and 77:

Pomerantz et al. and Saito substantially teach the method/system as applied to claims 1, 26, 51, and 72. Furthermore, Pomerantz et al. teach the method/system wherein the graphics device is a memory device (col. 5, line 66 – col. 6, line 2).

As per claims 7, 32, 57, and 78:

Pomerantz et al. and Saito substantially teach the method/system as applied to claims 1, 26, 51, and 72. Furthermore, Saito teaches the method/system wherein the graphics device is a screen device (col. 4, lines 35-44).

As per claims 8, 33, 58, and 79:

Pomerantz et al. and Saito substantially teach the method/system as applied to claims 1, 26, 51, and 72. Furthermore, Saito teaches the method/system wherein the graphics device is a graphics port[2] (col. 8, lines 19-36). Although there is no explicit reference made to a graphics port, the elements referred to in the detailed description use ports to transfer graphics, thus it is identical to there being a graphics port.

As per claims 12 and 37:

Pomerantz et al. and Saito substantially teach the method/system as applied to claims 1 and 26. Furthermore, Pomerantz et al. teach the method wherein the content and said encrypting comprises padding encrypted text so that identical words have distinct encrypted representations (col. 7, lines 1-41). Although the term padding is not used, the definition of padding[3] suggests that it is inherent.

As per claims 14, 39, 60, and 81:

---

[2] The definition of port as pasted from www.netlingo.com is as follows:

> **Port** - Commonly known as the place where information goes into and out of a computer, or both. For example, the serial port on a personal computer is where a modem or printer is connected.
>
> On the Internet, "port" often refers to a number that is shown in a URL, following a colon right after the domain name. Every service on an Internet server "listens" on a particular port number. Most of these services have standard port numbers. Web servers normally listen on port 80, and the standard Gopher port is 70. (Services can also listen on nonstandard ports, in which case the port number must be specified in a URL when the server is accessed.)

[3] According to the Hacking Lexicon dictionary online, the definition of padding is as follows:

> **Padding** - Padding is the process of adding unused data to the end of a message in order to make it conform to a certain length. For example, block-ciphers often work on blocks that are 64-bits (8-bytes) long. Therefore, if you have a message that is 77-bytes long, you will need to "pad" it with an extra 3-bytes to make it an even 80-bytes in size (10-blocks).
> **Key point:** Padding is a regular feature of all crypto algorithms, including hashing and encryption. Some algorithms have been broken due to poor choices for padding. Most importantly, however, the size of the message can often reveal details about its contents. For example, let's assume a protocol whereby somebody accepts something with a simple message of "yes", but when it declines, it says "no" along with a reason why it was rejected. Therefore, even though the messages are encrypted, the "yes" will be a short message but the "no" will be a long message.

Pomerantz et al. and Saito substantially teach the method/system as applied to claims 1, 26, 51, and 72. Furthermore, Saito teaches the method/system wherein the output data is a raster output data (col. 4, lines 35-44). Although the term "raster output" is not explicitly used, a CRT[4] is used as the display device, hence it is identical to that of a "raster output."

As per claims 15 and 40:

Pomerantz et al. and Saito substantially teach the method/system as applied to claims 1 and 26. Furthermore, Pomerantz et al. teach the method/system wherein said identifying, and said modifying are performed by a server computer, and wherein, said rendering, and said displaying are performed by a client computer connected to the server computer over a network (col. 10, lines 55-59).

As per claims 16:

Pomerantz et al. and Saito substantially teach the method as applied to claim 1. Furthermore, Saito teaches the method/system occurring within a patched operating system function for outputting content (col. 5, lines 28-35).

As per claims 17, 42, 62, and 83:

---

[4] The definition of Cathode Ray Tube (CRT) from the Free Online Dictionary of Computing is as pasted below:

> **CRT** - An electrical device for displaying images by exciting phosphor dots with a scanned electron beam. CRTs are found in computer VDUs and monitors, televisions and oscilloscopes. The first commercially practical CRT was perfected on 29 January 1901 by Allen B DuMont.
>
> A large glass envelope containing a negative electrode (the cathode) emits electrons (formerly called "cathode rays") when heated, as in a vacuum tube. The electrons are accelerated across a large voltage gradient toward the flat surface of the tube (the screen) which is covered with phosphor. When an electron strikes the phosphor, light is emitted. The electron beam is deflected by electromagnetic coils around the outside of the tube so that it scans across the screen, usually in horizontal stripes. This scan pattern is known as a raster. By controlling the current in the beam, the brightness at any particular point (roughly a "pixel") can be varied.

Pomerantz et al. and Saito substantially teach the method/system as applied to claims 16, 26, 51, and 82 above. Furthermore, Saito teaches the method/system wherein the patched operating system function is a TextOut function (col. 4, lines 45-57).

As per claims 18, 43, 63, and 84:

Pomerantz et al. and Saito substantially teach the method/system as applied to claims 16, 41, 51, and 82 above. Furthermore, Saito teaches the method/system wherein the patched operating system function is a type of DrawText function (col. 4, lines 45-57).

As per claims 25, 50, 70, and 91:

Pomerantz et al. and Saito substantially teach the method/system as applied to claims 24, 49, 69, and 90. Furthermore, Saito teaches the method/system wherein at least one of the spatial characteristics is computed by a GetTextExtent function (col. 4, lines 45-57).

As per claims 51, 72, and 172:

Pomerantz et al. teach a method/system/computer readable medium with stored program code, wherein the page contains a portion of encrypted text, comprising: rendering[5] the page into a graphics device comprising: determining spatial characteristics form the decrypted text and generating a layout for the page based on the special characteristics, wherein the spatial characteristics include at least one of (a) positions of characters, (b) heights of characters, (c) widths of characters, (d) widths of words, (e) shapes of characters, (f) spacings between characters, (g) spacings between words, (h) spacings between lines, (i) numbers of characters per line, (j) numbers of words per line, (k) page margins, and (l) paragraph indentations (col. 8, line 8-20; col. 8, line 34-49; and Fig. 2A); and converting the page into output data for a graphics

[5] See footnote (1) on page 4

device (col. 6, lines 28-34); and displaying at least a portion of the output data (col. 6, lines 28-34).

Not explicitly disclosed is dynamically generating a display layout and decrypting the portion of encrypted text within a patched operating system function to produce decrypted text. However, Saito teaches rendering a display to a web page, text that is to be output replaces a certain portion of page and wherein web pages are known to dynamically generate a display layout and using a patched operating system function (col. 5, lines 28-39 and col. 6, line 1-7). Therefore, it would have been obvious to a person in the art at the time the invention was made to modify the method disclosed in Pomerantz et al. for the rendering function to include dynamically generating a display layout for a web page and replacing the text input portion, which would be the on-screen section selected for a cut/copy command (taught by Pomerantz et al.), with another text output portion which is the decrypted form of the text retrieved by the patched operating system function such as a TextOut() or ExtTextOut() function (taught by Saito) and then displaying the decrypted text to the screen. This modification would have been obvious because a person having ordinary skill in the art, at the time the invention was made, would have been motivated to do so since Saito teaches that rendering text in the format described allows for convenient text extraction and replacing any arbitrary word(s) on a screen and results in dynamically generating the display layout of the text as well in col. 1, lines 59-61. As per claims 69 and 90:

Pomerantz et al. and Saito substantially teach the method/system as applied to claims 68 and 89. Furthermore, Saito teaches the method/system wherein the patched operating system function is adapted to determine widths of character strings (col. 6, lines 1-7).

As per claim 71:

Pomerantz et al. and Saito substantially teach the method/system as applied to claims 51. Furthermore, Pomerantz et al. teach the method further comprising receiving the page having a portion of encrypted content from a server computer (col. 10, lines 55-59).

As per claim 88:

Pomerantz et al. and Saito substantially teach the system as applied to claim 72. Furthermore, Pomerantz et al. teach the method/system, wherein said dynamically generating comprises calculating widths of character strings (col. 8, line 34-49).

As per claim 89:

Pomerantz et al. and Saito substantially teach the system as applied to claim 88. Furthermore, Pomerantz teach the method/system, wherein said dynamically generating comprises decrypting encrypted text strings (col. 8, line 34-49).

As per claim 92:

Pomerantz et al. and Saito substantially teach the method/system as applied to claims 72. Furthermore, Pomerantz et al. teach a system further comprising a network connector and a receiver receiving the page having a portion of encrypted content from a server computer via said network connector (col. 10, lines 55-59).

As per claims 115, 124, and 173:

Pomerantz et al. substantially teaches a method/system/computer readable medium with stored program code comprising formatting the page by generating a layout for the page based on special characteristics including at least one of (a) positions of characters, (b) heights of characters, (c) widths of characters, (d) widths of words, (e) shapes of characters, (f) spacings

between characters, (g) spacings between words, (h) spacings between lines, (i) numbers of characters per line, (j) numbers of words per line, (k) page margins, and (l) paragraph indentations (col. 8, line 8-20; col. 8, line 34-49; and Fig. 2A); rendering the page into a graphics device according to the page layout, said rendering comprising replacing the portion of encrypted text with the decrypted text (col. 7, lines 1-9), converting the page into output data for a graphics device (col. 6, lines 28-34), and writing the output data into the graphics device (col. 6, lines 28-34).

Not explicitly disclosed is dynamically generating a display layout and decrypting the portion of encrypted text within a patched operating system function to produce decrypted text. However, Saito teaches rendering a display to a web page, text that is to be output replaces a certain portion of page and wherein web pages are known to dynamically generate a display layout and using a patched operating system function (col. 5, lines 28-39 and col. 6, line 1-7). Therefore, it would have been obvious to a person in the art at the time the invention was made to modify the method disclosed in Pomerantz et al. for the rendering function to include dynamically generating a display layout for a web page and replacing the text input portion, which would be the on-screen section selected for a cut/copy command (taught by Pomerantz et al.), with another text output portion which is the decrypted form of the text retrieved by the patched operating system function such as TextOut() or ExtTextOut() function (taught by Saito) and then displaying the decrypted text to the screen. This modification would have been obvious because a person having ordinary skill in the art, at the time the invention was made, would have been motivated to do so since Saito teaches that rendering text in the format described allows for

convenient text extraction and replacing any arbitrary word(s) on a screen and results in dynamically generating the display layout of the text as well in col. 1, lines 59-61.

As per claims 116 and 125:

Pomerantz et al. and Saito substantially teach the method/system as applied to claims 115 and 124. Furthermore, Pomerantz teaches wherein the portion of encrypted text includes encrypted words of text having the same respective widths as corresponding words of text included in the decrypted text (col. 8, lines 8-20 and Fig. 2A-2B).

As per claims 117 and 126:

Pomerantz et al. and Saito substantially teach the method/system as applied to claims 115 and 124. Furthermore, Pomerantz teaches the method/system wherein the output data is a raster output data (col. 4, lines 35-44).

As per claims 118 and 127:

Pomerantz et al. and Saito substantially teach the method/system of replacing the portion of encrypted text with the portion of decrypted text. Furthermore, Saito teach the method occurring within a patched operating system function for converting text into the graphics output (col. 5, lines 28-35).

As per claims 119 and 128:

Pomerantz et al. and Saito substantially teach the method/system wherein said replacing the first portion of text with a second portion of text occurs within a patched operating system function for converting text into the graphics output, as applied to claims 118 and 127 above. Furthermore, Saito teaches the method/system wherein the patched operating system function is a TextOut function (col. 6, lines 45-63).

As per claims 120 and 129:

Pomerantz et al. and Saito substantially teach the method/system wherein said replacing the first portion of text with a second portion of text occurs within a patched operating system function for converting text into the graphics output, as applied to claims 118 and 127 above. Furthermore, Saito teaches the method/system wherein the patched operating system function is a type of DrawText function (col. 6, lines 45-63).

As per claims 121 and 130:

Pomerantz et al. and Saito substantially teach the method/system as applied to claims 115 and 124. Furthermore, Pomerantz et al. teach the method wherein said formatting comprises replacing encrypted text strings with decrypted text strings and calculating widths of the decrypted text portion based on selected font types and font sizes (col. 8, line 1-34).

As per claims 122 and 131:

Pomerantz et al. and Saito substantially teach the method/system wherein said formatting comprises replacing encrypted text strings with decrypted text strings and calculating widths of the decrypted text strings based on selected font types and font sizes as applied to claims 121 and 130 above. Furthermore, Saito teach the method/system occurring within a patched operating system function for determining widths of character strings (col. 6, lines 1-7).

As per claims 123 and 132:

Pomerantz et al. and Saito substantially teach the method/system wherein replacing first text strings with second text strings occurs within a patched operating system function for determining widths of characters as applied to claims 122 and 131 above. Furthermore, Saito et

al. teach the method/system wherein the patched operating system function is a GetTextExtent function (col. 6, lines 45-63).

As per claims 141-142 and 174:

Pomerantz et al. substantially teach a method for protecting text within a page displayed by a computer comprising: encrypting the plurality of encrypted text strings (col. 7, lines 1-9); replacing the plurality of encrypted text strings within the page with the plurality of decrypted text strings (col. 7, lines 1-9; col. 8, lines 1-34; and Figs. 2A-2B); determining spatial characteristics of the decrypted text strings (col. 8, line 8-20; col. 8, line 34-49; and Fig. 2A); and deriving a layout for the page based on spatial characteristics of the decrypted text strings (col. 7, lines 52-63).

Not explicitly disclosed is performing the above steps using a patched operating system function. However, Saito teaches that when rendering a display to a web page, an operating system function is used to replace words/text strings with different text, where the operating system has a function call to the text that is to be output (col. 5, lines 28-39 and col. 6, line 1-7). Therefore, it would have been obvious to a person in the art at the time the invention was made to modify the method disclosed in Pomerantz et al. for the rendering function to include dynamically generating a display layout for a web site and replacing the text input portion, which would be the on-screen section selected for a cut/copy command (taught by Pomerantz et al.), with another text output portion which is the decrypted form of the text retrieved by the patched operating system function such as TextOut() or ExtTextOut() function (taught by Saito) and then displaying the decrypted text to the screen. This modification would have been obvious because a person having ordinary skill in the art, at the time the invention was made, would have been

motivated to do so since Saito teaches that rendering text in the format described allows for convenient text extraction and replacing any arbitrary word(s) on a screen in col. 1, lines 59-61.

V. Claims 4, 29, 54, and 75 are rejected under 35 U.S.C. 103(a) as being unpatentable over Pomerantz et al. US Patent No. 6,178,243 and Saito, US Patent No. 5,900,005 as applied to claims 2, 27, 52, 73, 94, and 105 above, and further in view of the definition of XML, found at netlingo.com.

As per claims 4, 29, 54, and 75:

Pomerantz et al. and Saito substantially teach the method/system as applied to claims 2, 27, 52, and 73. Not explicitly disclosed is the method/system wherein the web page is an XML page. However, Howard et al. teach the method/system wherein the web page is an HTML page. Therefore, it would have been obvious to a person in the art at the time the invention was made to modify the method disclosed in Pomerantz et al. to incorporate the web page as an XML page. This modification would have been obvious because a person having ordinary skill in the art, at the time the invention was made, would have been motivated to do so since it is suggested by the definition of XML as found on www.netlingo.com and pasted below:

> A programming language/specification developed by the W3C. XML is a pared-down version of SGML, designed especially for Web documents. It enables Web authors and Web developers to create their own customized tags to provide functionality not available with HTML. For example, XML supports links that point to multiple documents (as opposed to HTML links, which can reference just one destination each). XML provides a powerful set of tools for developing a new generation of Web applications, including tools like database exchange, distribution of processing to clients, multiple views of data, intelligent agents, management of document collections, and so on.

VI. Claims 9-11 and 34-36 are rejected under 35 U.S.C. 103(a) as being unpatentable over Pomerantz et al. US Patent No. 6,178,243 and Saito, US Patent No. 5,900,005 as applied to claims 1 and 26 above, and further in view of Bloomberg United States Patent No. 5,761,686.

As per claim 9 and 34:

Pomerantz et al. and Saito substantially teach the method/system as applied to claims 1 and 26. Not explicitly disclosed is the method/system wherein said encrypting is based on encoding of characters. However, Bloomberg teaches the method/system wherein said encrypting is based on encoding of characters. Therefore, it would have been obvious to a person in the art at the time the invention was made to modify the method/system disclosed in Pomerantz to carry out the encryption based on an encoding of characters. This modification would have been obvious because a person having ordinary skill in the art, at the time the invention was made, would have been motivated to do so since it is suggested by Bloomberg in col. 8, lines 4-8.

As per claim 10 and 35:

Pomerantz et al. and Saito substantially teach the method/system as applied to claims 1 and 26. Not explicitly disclosed is the method/system wherein said encrypting is based on encoding of words. However, Bloomberg teaches the method/system wherein said encrypting is based on encoding of words. Therefore, it would have been obvious to a person in the art at the time the invention was made to modify the method/system disclosed in Pomerantz et al. to carry out the encryption based on an encoding of words. This modification would have been obvious because a person having ordinary skill in the art, at the time the invention was made, would have been motivated to do so since it is suggested by Bloomberg in col. 12, lines 22-29.

As per claim 11 and 36:

Pomerantz et al. and Saito substantially teach the method/system as applied to claims 1 and 26. Not explicitly disclosed is the method/system wherein said encrypting comprises adding

leading and trailing characters to flag encrypted text. However, Bloomberg teaches the method/system wherein said encrypting comprises adding leading and trailing characters to flag encrypted text. Therefore, it would have been obvious to a person in the art at the time the invention was made to modify the method/system disclosed in Pomerantz et al. to carry out the encryption and adding leading and trailing characters to flag encrypted text. This modification would have been obvious because a person having ordinary skill in the art, at the time the invention was made, would have been motivated to do so since it is suggested by Bloomberg in col. 13, lines 22-26.

References Cited, Not Used:

The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

(1) US Patent No. 6,282,653
(2) US Patent No. 6,052,780
(3) US Patent No. 5,822,432
(4) US Pub. No. 2002/0188570
(5) US Pub No. 2002/0021807
(6) US Patent No. 5,983,227 – Specifically addresses the benefits of dynamic page generation.

The previously cited references are relevant due to the manner in which the invention is claimed.

Conclusion

THIS ACTION IS MADE FINAL. Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period

will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Nadia Khoshnoodi whose telephone number is (571) 272-3825. The examiner can normally be reached on M-F: 8:00-4:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Emmanuel Moise can be reached on (571) 272-3865. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

/Nadia Khoshnoodi/
Examiner, Art Unit 2437
3/14/2009

NK

/Emmanuel L. Moise/
Supervisory Patent Examiner, Art Unit 2437